Alzheimer’s Finder

FINAL REPORT

Apostolos Vailakis | Fragoulis Logothetis | LAB41130792

Project Overview

A common disease of the mind that occurs on many elder people is the Alzheimer's disease. It is the cause of 60% to 70% of cases of dementia. The most common early symptom is difficulty in remembering recent events (short-term memory loss). As the disease advances, symptoms can include problems with language, disorientation (including easily getting lost), mood swings, loss of motivation, not managing self-care, and behavioral issues.

Our project's main objective was to provide a method for locating easily lost relatives, who suffering from this disease choose to wonder outside of their home premises without informing their close ones. This has lead us to the development of a GPS tracking device capable of informing the user about the patient's position using the GSM standard as a means of communication.

The project is also expected to function for a reasonable amount of time since it is a device for emergency situations. This also enables the device to stay on-line for grater time periods which increases the odds of full GPS and GSM coverage.

Real-Life Scenario

Patients who suffer from Alzheimer's disease usually decide to wonder aimlessly in the streets. Having that in mind we decided to use a GPS module as our locating system. Cases where neither the GPS or GSM signals can be found are usually inside buildings and out of harm's way. The device is also equipped with two lithium-ion batteries (type 18650) providing enough power for the device to function properly for roughly a week, ensuring that even if the patient is not found immediately, his relatives will have the option of GPS tracking for days to come, increasing significantly their chances. The user is able to communicate with the remote device using an app on his cellphone which makes the whole operation much easier. The corresponding message from the device contains not only the patients position in a form of URL for immediate redirection on a google maps location but also the battery level helping the user decide whether they'd like to conserve battery by lowering the frequency of their communication. Why this conserves energy will be explained later on the hardware section.

Hardware

The device consists of four main components. A battery pack, a Teensy 3.2 microcontroller, a GSM sim900 module and finally a GY-NEO6MV2 module.

Battery Pack:

The battery pack consists of two li-ion batteries (type 18650). These are commonly used in laptop battery packs and are designed to cut-off their power output when their voltage has reached a certain level. This is essential since batteries of this technology can behave destructively when discharged below a certain point. The batteries a connected in series giving a total of 8,4 Volts when fully charged and a total of 5,6 Volts before their internal protection circuit disconnects them.

Main Microcontroller:

The project utilizes a Teensy 3.2 microcontroller although its much cheaper brother, the Teensy LC, would be more than enough for this job. The microcontroller was chosen for its already built-in UART buses which are utilized for communicating with the GPS and GSM modules and for the already built-in USB port which in turn is utilized as a programming and debugging port.

GSM Module:

The chosen GPS module was a SIM900 chipset in a shield board designed originally for use with a more common Arduino UNO. This was very helpful as the module had its own voltage regulator which was used to power the rest parts of the device. The power consumption of the extra modules connected to the GSMs is small enough not to excessively overuse its power regulator since the GSM module is lightly used.

GPS Module:

For GPS tracking a GY-NEO6MV2 module was used which comes with an easy to use library for communication, making the whole development process a bit easier.

Power Management:

In order to conserve energy and make the battery last al lot longer the device is capable of disabling completely the GPS module, reenabling it only when it is necessary for location tracking (i.e. when the user asks for the device's location). This technique slows down the response time of the device since the GPS tracker needs about 25 seconds after a cold-start to orient itself. Though unfortunate this drawback is a small price to pay for the enormous amount of battery conserved (an average of 50mA is required by the GPS module when in use). Another key modification done to further lower the power consumption was lowering the clock speed of the main microcontroller. This enabled the microcontroller to work at around 11mA. Finally, the GSM module was put in “low power mode” in which it significantly lowers its power consumption at roughly 4mA without really compromising the performance of the device, since the GSM module is primarily used as a “listening” device, something which is hardly affected by the “low power mode”

Power Scenario:

The above Current measurements where done before the voltage regulators, so the power lost by heat dissipation from the voltage regulators is taken into account. The device is mainly fluctuating between two states, the Active State and the Passive State.

Passive State:

> In the Passive State the only active modules are the GSM module and the Teensy 3.2 consuming a total of 15mA

Active State:

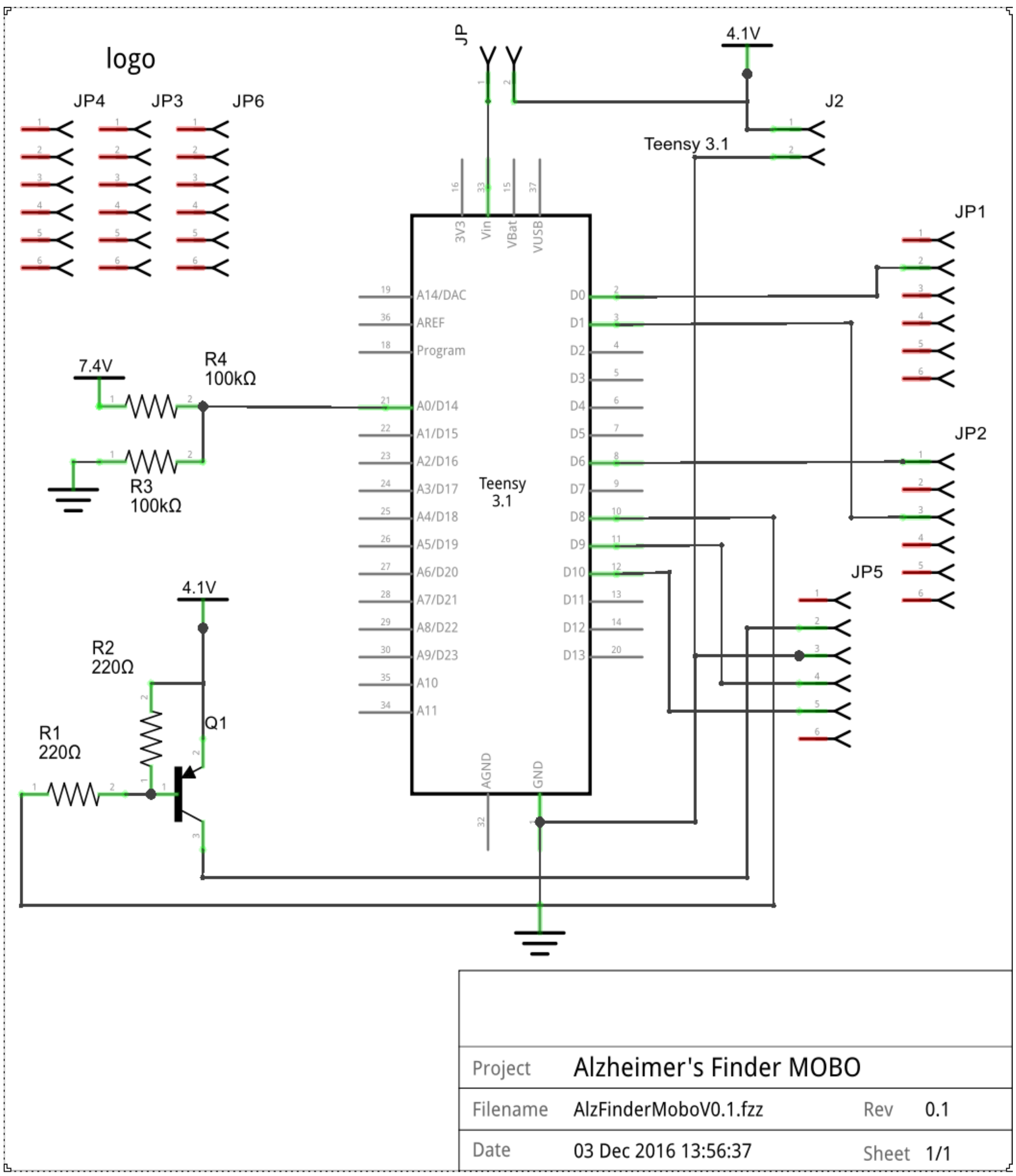
> In the Active state an additional 50 mA are consumed by the GPS module draining a total of 65 mA from the Battery

Each time the device is asked to perform a GPS location check and report back to the user, it changes from the Passive state, to the Active state, in which it stays for about 30 seconds (25 seconds for the GPS module to orient itself + 5 seconds for the rest procedures). For simplicity reasons, we will assume 40 second Active State cycle which is more than enough to accommodate for the extra Current needed by the GSM module to exit Power Save Mode and send a message to the user.

If we assume that in an emergency situation a 1000 Active State cycles occur (something quite overstretched just to prove o point) and that the average voltage provided by the batteries is 7 Volts which means 2300 mAh before they are fully depleted. We end up with 11 hours of the device working in the Active State, and another roughly 109 hours in Passive State before the batteries are fully depleted (11 * 65 + 109 * 15 = 2350). Combined the device provides easily 5 days of continuous usage even when its position is acquired by the user very frequently. If the user decides to track the device less frequently, the battery life can be stretched to roughly 9 days.

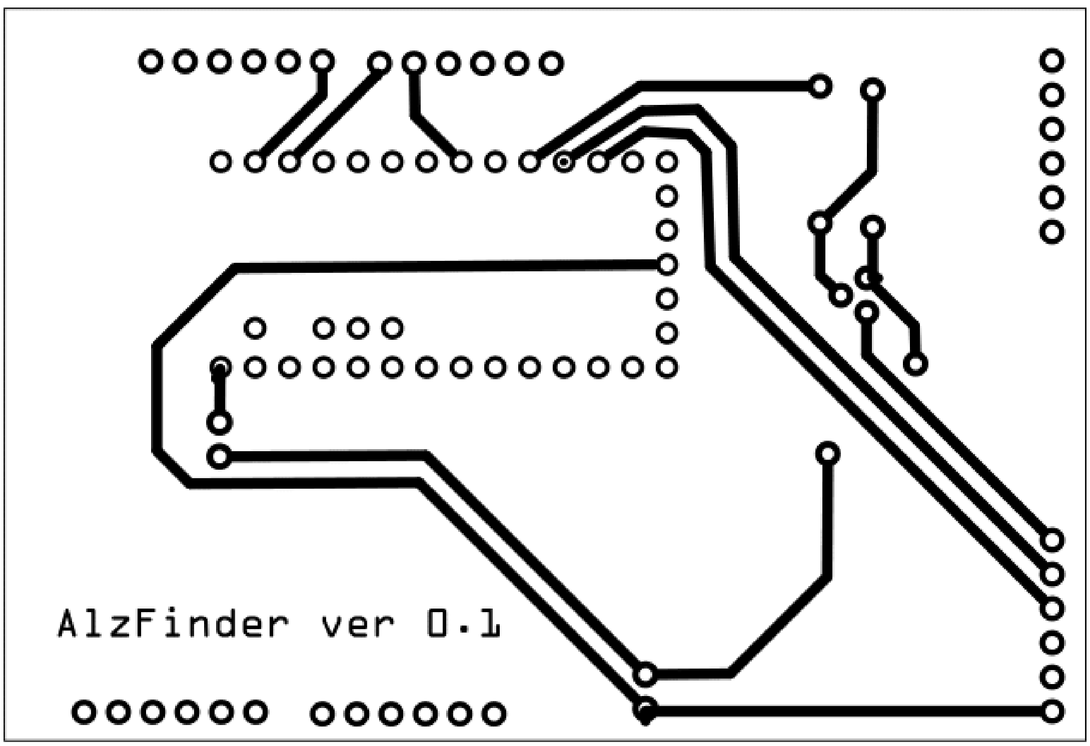
Schematic:

The purpose of this circuit is not only to connect correctly the three modules used for this project, but also to enable the Teensy microcontroller to power-up the GPS module only when it's needed, thus conserving precious energy. The microcontroller and GPS module share the power coming from the GSM board which contains a rather beefy Voltage regulator. A voltage divider is used before the GSM's voltage regulator so the current state of the batteries can be monitored by the microcontroller using one of its analog inputs. This schematic contains only the Teensy microcontroller and no other module, since all the other modules are connected to the main board through external pins.



PCB Design:

The PCB design is based on the outline of a standard "Arduino Uno", the reason for that is the GSM module since it was originally designed to be used as a "shield" in conjunction with an Arduino Uno. This helps the PCB and the rest modules to be connected all together in a tight-spaced and sturdy manner. An L shaped connection has been formed with another board in the far right of the main PCB for the GPS module to be connected. This does not show in the main PCB but is the reason why some extra pins are used for structural support. This L shaped connection can be seen in pictures provided further down this document and was created so that the GPS module will be facing upwards when the hole device is mounted to a patient.

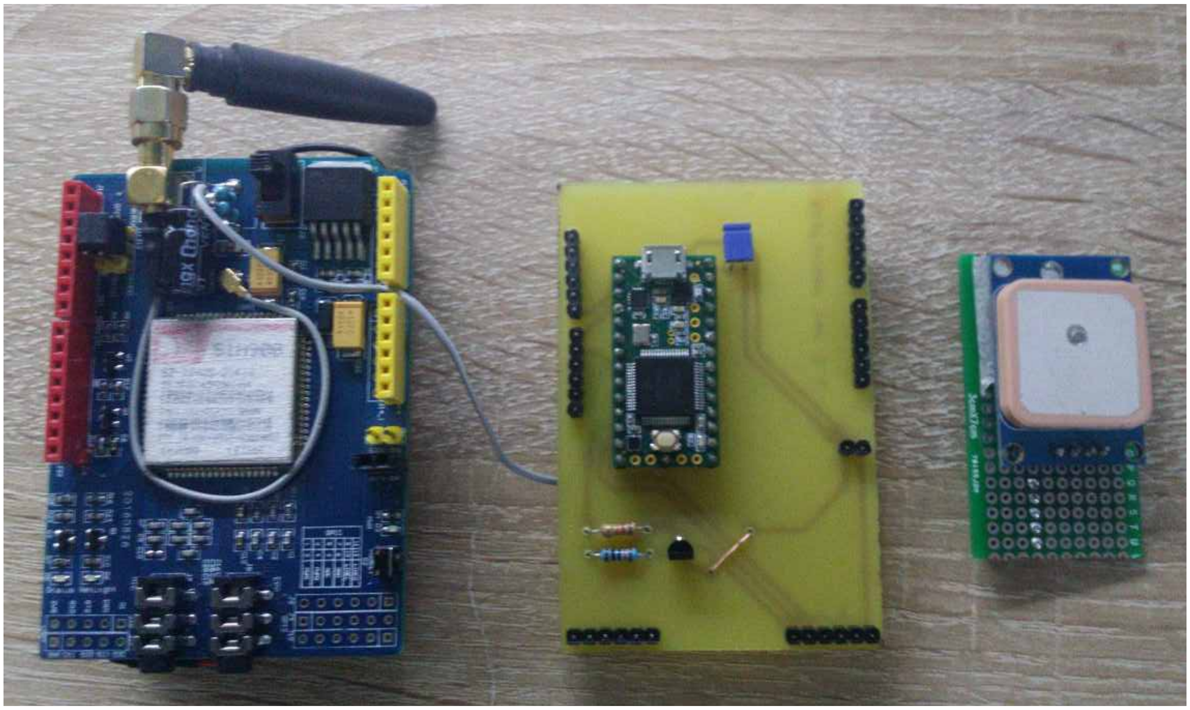


Battery Mounts:

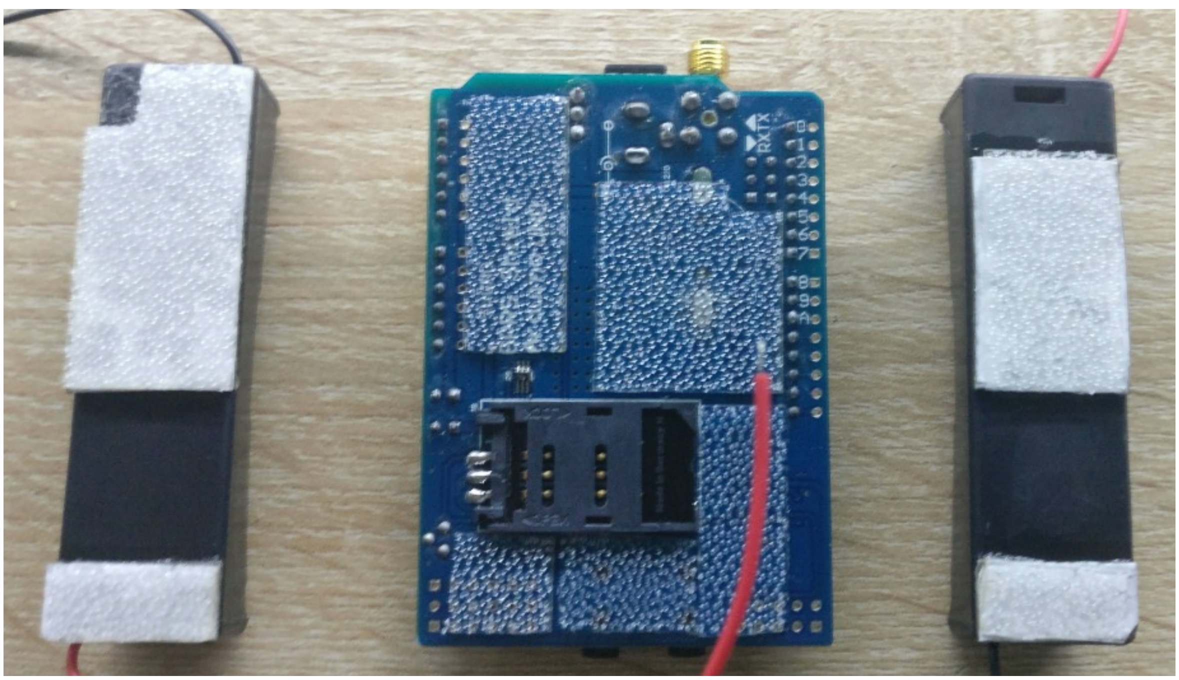
Two battery holders are used for the batteries so that they can be extracted from the device and charged externally. Although this charging method is requiring more effort from the user, it enables the devise be quickly refitted with freshly charged batteries, ensuring it will always be mounted on the patient and function. The battery holders are attached to the GSM module using some "3M Dual Lock" (A more industrial and sturdier version of the commonly used "Velcro") so they can be removed for a SIM card to be inserted.

Build Stages and Final Product:

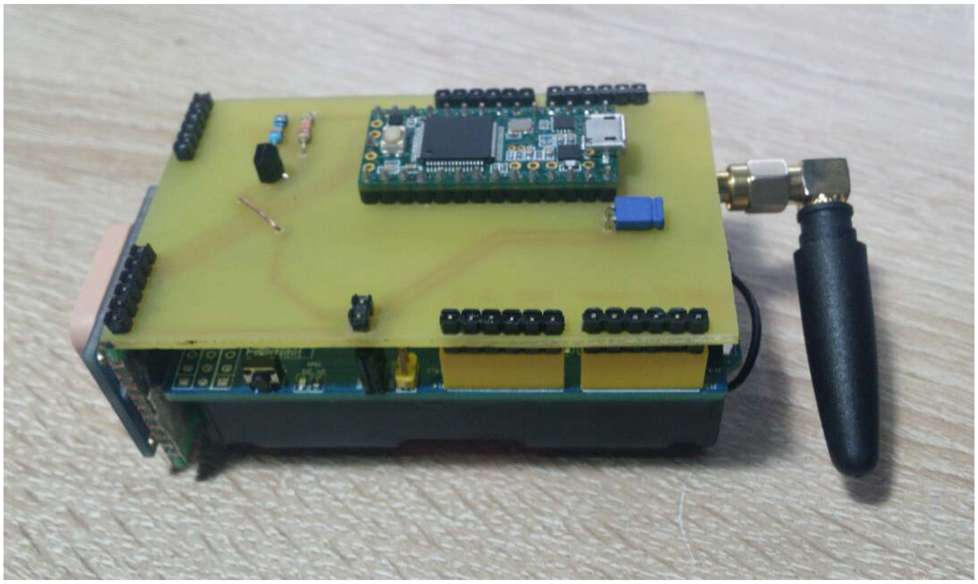
Unconnected Modules:



Battery Mounting System



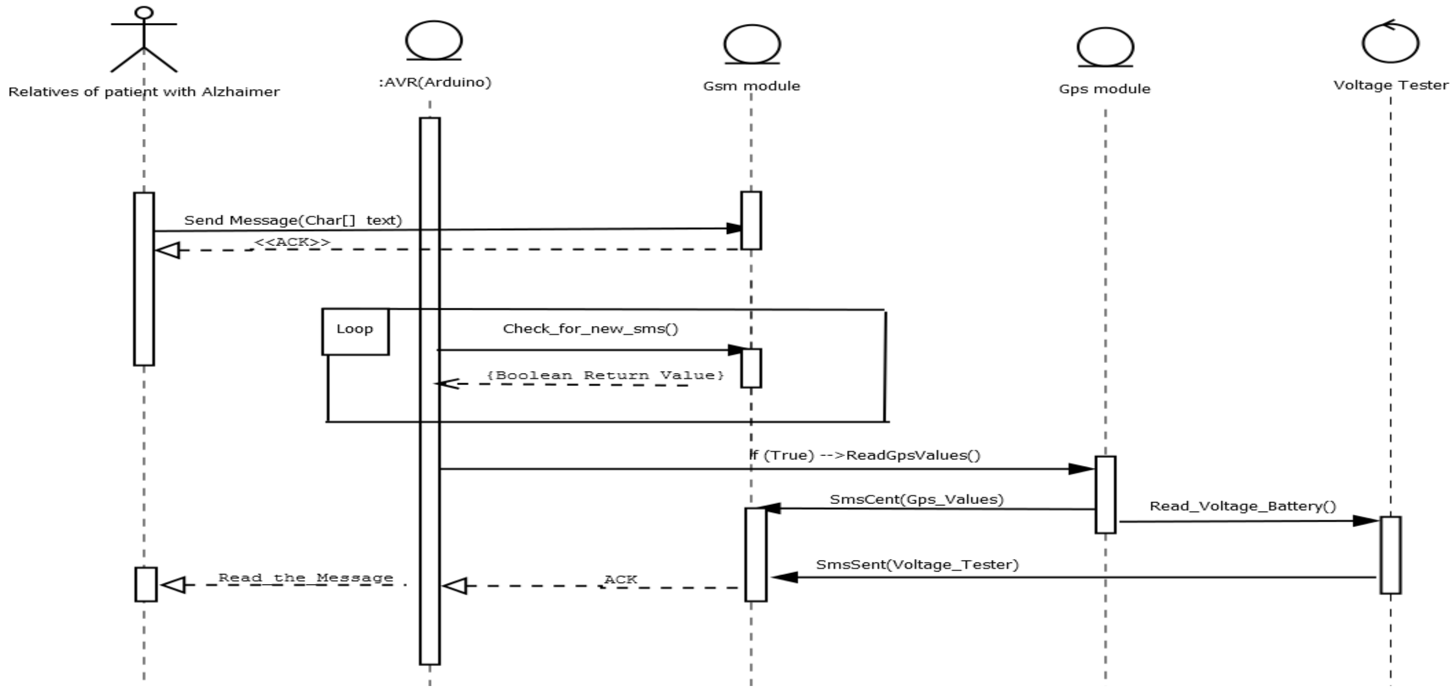
Connected Modules:



Final Product:

Software:

The Basic function of the device can be seen in the following Sequence Diagram:



Programming part could be divided into two pieces. ARM programming and Android programming .

ARM PROCCESSOR PROGRAMMING

The aim of this thesis is to present an embedded intelligence control system based on proposed drivers and applied to the Teensy platform. We used the open source Gps Library to achieve the best regulation of the Gps Module therefore, the most important goal was to lower the pwer consumption of the Gsm module. To achieve the above goal, a real time driver based on AT COMMAND datasheet is developed, in order to be implemented on the Teensy microcontroller board.

ANDROID APPLICATION

The Android app is built according to the best practice and offers a wide variety of options to the end user. We used the Android Studio Platform and a variety of techniques to build a high-level program, in order to run as fast as possible on a cheap processor .Some of the implemented tricks are:

-> Loop fusion →Loop nest splitting

→Loop unrolling →Look-up Branches (ARM compiler)

Programming an Arduino Application in Android 6.0 Marshmallow turns out to be complicated, as marshmallow needs dynamic agreement for I/0's to ROM, to send an sms, etc.

Future Improvements:

Possible Hardware Improvements:

For the device to be even more easy to carry two upgrades can be implemented.

- Replace type 18650 Li-Ion Batteries with Two Li-ion Cells providing the same power but in a more compatible shape, saving space and making the whole device smaller. In addition, the lack of battery holders provides enough extra room for a Li-Ion battery controller, introducing USB charging capabilities.
- Replace SIM900 module with a similar SIM800 module, which includes a GPS module. This eliminates the rather large GPS module that is currently used.
- Design a multilayer PCB containing a Cortex-M0+ and the SIM800 chipset.

These refinements, although much more time consuming to design, can actually half the overall size of the device.

Possible Software Improvements:

For a much more reliable system, improvements can be made to increase the device's chances of correctly locating itself and the patient. Such improvements include remembering the device's last known position and enabling tracking methods using only the GSM network. This way the device isn't stranded by the GPS network's lack of coverage in certain places.